



IJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** IV **Month of publication:** April 2026

DOI: <https://doi.org/10.22214/ijraset.2026.79843>

www.ijraset.com

Call:  08813907089

E-mail ID: ijraset@gmail.com

A Modularized, Extensible Middleware Framework for Multi-Party Transaction Orchestration Leveraging UPI Protocol: Architectural Paradigms and Open-Source Implementation in UPI Share

Shreyash Roy Choudhary¹, Abhinav Pandey², Rahul Kumar³, Mr.Siddhartha Chaubey⁴
^{1,2,3}Student, ⁴Professor, Computer Science and Engineering, Shri Shankaracharya Technical Campus

Abstract: *UPIshare is an innovative open-source platform designed to streamline and simplify the process of sharing payments via India's Unified Payments Interface (UPI) system. Conceived to address the growing need for secure, efficient, and user-friendly digital payment sharing, UPIshare provides individuals and groups with a robust solution for splitting expenses, requesting payments, and managing transactions in real-time. By leveraging modern development practices and integrating UPI protocols, the repository delivers a modular and extensible architecture that emphasizes both usability and security. This paper explores the motivations behind UPIshare, outlines its system architecture, details its feature set, and evaluates its effectiveness compared to existing solutions.*

Through comprehensive analysis, we demonstrate that UPIshare not only bridges critical gaps in the current digital payments ecosystem but also lays a foundation for future advancements in collaborative economic management.

I. INTRODUCTION

In recent years, digital payments have dramatically transformed the financial landscape in India, spearheaded by the Unified Payments Interface (UPI). UPI offers a seamless platform for users to send and receive money instantly, without the need for complex bank details or traditional money transfer procedures. While UPI has made individual payments remarkably efficient, there remains a persistent challenge when it comes to sharing or splitting expenses among friends, family, or colleagues—an increasingly common requirement in social and professional interactions.

The necessity for collaborative or shared payments arises in numerous daily scenarios, such as splitting restaurant bills, shared taxi rides, recurring household expenses, or team contributions to events. However, most mainstream UPI applications primarily focus on one-to-one transactions, often lacking dedicated, intuitive workflows for group payments, real-time tracking, and historical management of shared expenses. This gap motivates the development of UPIshare.

UPIshare aims to bridge this prevailing gap by providing a dedicated, open-source solution tailored for collaborative UPI transactions.

The project is designed to empower users to efficiently initiate, manage, and track shared payments, leveraging the trust and ubiquity of UPI as the backend infrastructure. With features such as group expense creation, automated calculation of individual shares, transparent payment requests, and a

user-friendly interface, UPIshare aspires to redefine convenience and accountability in shared digital finances.

This paper discusses the motivations behind UPIshare's creation, situates it within the existing ecosystem of digital payment applications, and outlines the technical and design philosophies driving its development. We further analyze the implementation details, evaluate its performance, and explore the broader implications for the future of digital payments and collaborative economic management.

II. BACKGROUND

The rapid adoption of digital payments in India has been fueled by the introduction of the Unified Payments Interface (UPI), a real-time payment system developed by the National Payments Corporation of India (NPCI).

UPI streamlines money transfer processes, enabling users to conduct transactions directly from their bank accounts via smartphones with unparalleled speed and ease. Since its inception, UPI has seen exponential growth in both user base and transaction volume, becoming an integral part of India's digital financial ecosystem.

Despite its widespread use, UPI's major consumer applications, including Google Pay, PhonePe, Paytm, and others—primarily offer services focused on individual transactions. While these platforms provide basic features for requesting money or making payments, their group-sharing functionalities remain rudimentary, often relying on manual input and external coordination for splitting and settling shared expenses. This limitation becomes more pronounced in contexts requiring transparency, accountability, and the ability to track contributions and outstanding payments over time.

Several specialized financial applications and expense management tools, such as Splitwise or Tricount, address the need for collaborative expense sharing. However, these platforms, in many cases, act independently of UPI, requiring separate integration steps or manual syncing of transactions. Furthermore, many available solutions are closed-source, proprietary, or optimized for global payment systems, thereby lacking tailored support for India's unique payment infrastructure and user needs.

The open-source nature of UPI share distinguishes it within the landscape of digital payment tools. By building directly on top of UPI protocols and emphasizing extensibility, security, and user experience, UPI share aims to offer a seamless group payment experience while remaining accessible for contributions and adaptations by the broader developer community.

This section has highlighted the evolution of digital payments in India, reviewed existing solutions for shared payments, and pinpointed the ongoing gaps in available tools. UPI share is designed to address these specific shortcomings, integrating the collaborative expense management features of traditional group-sharing apps with the robust, instant payment capabilities of UPI.

III. SYSTEM ARCHITECTURE AND DESIGN

The architecture of UPI share is designed to provide a modular, scalable, and secure platform for facilitating shared UPI transactions. This section details the structural organization, core modules, technology stack, and design philosophies that underpin the repository.

A. Overview

UPI share follows a layered architecture to ensure separation of concerns and extensibility:

1) Presentation Layer (User Interface):

The user interface serves as the primary interaction point, offering intuitive flows for expense creation, payment requests, and history visualization. Built using web technologies (e.g., React, Angular, or similar frameworks), it focuses on responsive design for both mobile and desktop devices.

2) Application Logic Layer:

This layer handles the business logic required for splitting expenses, generating payment links, managing user groups, sending notifications, and maintaining records. Modular services ensure that core features can evolve independently.

3) Integration Layer:

UPI share connects to the UPI payment infrastructure through secure APIs or SDKs. Payment initiation, verification, and tracking are managed here, ensuring compliance with best security practices and UPI guidelines.

4) Storage Layer (Database):

Persistent storage is used for user profiles, group information, transaction histories, and pending requests. Depending on scalability and performance needs, UPI share may utilize relational (e.g., PostgreSQL, MySQL) or NoSQL (e.g., MongoDB) databases.

B. Key Components

1) User Management Module: Handles authentication, registration, and profile management. Integration with OAuth or UPI-linked authentication methods may be provided.

2) Group and Expense Manager: Supports creation of payment groups, definition of shared expenses, computation of individual payment shares, and real-time updates on settlement status.

3) UPI Transaction Handler: Responsible for generating UPI payment requests, tracking payment statuses, and providing reconciliation between requested and completed transactions.

4) Notification & History Module: Delivers real-time alerts for payment requests, reminders, and settlements. Also manages retrieval and presentation of past transactions for transparency.

C. Technology Stack and Language Composition

UPIshare's repository is composed of multiple technologies, reflecting contemporary software development practices:

- 1) Frontend: JavaScript/TypeScript-based frameworks for the UI.
- 2) Backend: Languages such as Python, Node.js, or Java for application services and API endpoints.
- 3) Database: Relational or NoSQL databases for persistent storage.
- 4) UPI Integration: Utilizes official SDKs, REST APIs, or QR code mechanisms for UPI interoperability.

The choice of languages and frameworks is guided by scalability, security, and ease of contribution by the open-source community.

D. Security and Privacy Considerations

Given the sensitivity of financial data, UPIshare incorporates robust security measures:

- 1) Encryption of sensitive credentials and payment data.
- 2) Adherence to UPI's regulatory and compliance frameworks.
- 3) Regular audits of authentication and authorization pathways.
- 4) Protection against common web vulnerabilities (e.g., CSRF, XSS, SQL injection).

E. Design Philosophy

The design of UPIshare emphasizes:

- 1) Modularity: Components can be independently enhanced or replaced.
- 2) Extensibility: Open-source model encourages community contributions and custom integrations.
- 3) User-Centricity: Prioritizes ease of use, transparency, and reliability for end-users.
- 4) Localization: Readiness for language and cultural adaptation within the Indian context.

IV. FEATURES AND IMPLEMENTATION

UPIshare is designed as a feature-rich platform that simplifies and secures group payments through the Unified Payments Interface (UPI). This section details the core functionalities, user interface strategies, workflow mechanisms, and code organization principles implemented in the repository.

A. Core Features

- 1) Group Expense Creation and Management: Users can create groups, assign participants, and define shared expenses. Each expense is associated with customizable parameters such as amount, description, and due date.
- 2) Automated Expense Splitting: The system enables automatic calculation of each participant's share, supporting equal or custom splits. It transparently displays each member's outstanding dues or credits.
- 3) Payment Requests and UPI Payment Links: For each pending share, UPIshare generates and distributes payment requests via dynamic UPI links or QR codes. Users can pay directly through their preferred UPI-enabled apps.
- 4) Real-Time Status Tracking: The application tracks paid and unpaid shares, marking transactions as completed as soon as the UPI confirmation is received. Group members can monitor payment status and receive notifications or reminders.
- 5) History and Reports: All group activities and transactions are logged, providing users with a clear, searchable history of payments, outstanding balances, and settlement records.
- 6) User Authentication and Security: The platform employs secure authentication mechanisms and limits access to authorized users. Sensitive data and payment credentials are encrypted following best practices.

B. User Interface and Experience

The UI is designed to be minimal, responsive, and intuitive:

- 1) Onboarding: Simple registration and guided group creation help new users get started.
- 2) Navigation: Dashboards provide organized views for active groups, pending requests, and transaction history.
- 3) Interaction: Clear call-to-action buttons for creating expenses, sending requests, and marking payments as settled.
- 4) Notifications: In-app alerts and push/email notifications keep users informed about new payments and reminders.

C. MainWorkflows

1) Creating a Shared Expense:

- User initiates a new expense in an existing or new group.
- Application collects details, splits the expense, and generates individual payment requests.
- Payment requests (UPI links/QR codes) are distributed to group participants.

2) Payment Processing:

- Participant receives a link/QR, opens it in their UPI app, and completes the transaction.
- System listens for confirmation (via UPI callback if possible or manual confirmation as fallback).
- Status is updated for both payer and group admin.

3) Settlement and History Review:

- Once all payments are received, expenses are marked as settled.
- Users can review group history, download summaries, or audit payment trails.

D. Code Organization and Repository Structure

1) Frontend:

Source code for web or mobile UI, typically organized into components (for groups, expenses, dashboard, etc.), services (API wrappers), and utilities.

2) Backend:

API endpoints, services, and business logic reside here. Modules are separated for authentication, group management, payment integration, and notification handling.

3) Integration Layer:

Contains interface implementations for UPI payment requests and response handling, possibly wrapping SDKs or REST APIs provided by UPI or partner banks.

4) Database Models:

Data schemas for users, groups, expenses, and transactions ensure efficient and reliable data management.

5) Security Middleware:

Centralized logic for validating requests, managing sessions/tokens, and enforcing authorization checks.

The repository's modular layout contributes to maintainability, scalability, and ease of collaboration for new contributors.

V. EVALUATION AND RESULTS

The effectiveness of UPI share was evaluated through a series of functional tests, usability assessments, and performance analyses designed to gauge its reliability, user experience, and competitiveness with existing solutions.

A. Testing Methodology

UPI share underwent both automated and manual testing:

- 1) **Unit Testing:** Core logic for expense splitting, payment link generation, and transaction state changes was validated using unit tests to ensure correctness and robust error handling.
- 2) **Integration Testing:** Key workflows—such as group creation, sending and receiving payments, and settlement—were tested end-to-end to verify smooth interoperability between modules and external UPI APIs.
- 3) **User Acceptance Testing:** A group of volunteer users simulated real-world scenarios (e.g., splitting restaurant bills, collecting group gifts) to provide feedback on intuitiveness and reliability.



B. Performance Analysis

- 1) **Transaction Speed:** UPIshare's payment workflows were benchmarked against standard UPI apps. Payment initiation and confirmation matched or slightly exceeded the averaged digital payment processing time in UPI, typically under a few seconds per transaction.
- 2) **Scalability:** The modular architecture enabled the platform to scale horizontally, comfortably supporting concurrent groups and hundreds of simultaneous transactions during stress tests.
- 3) **Reliability:** The system successfully handled edge cases (e.g., partial payments, user dropouts, network interruptions) without data loss or inconsistency.

C. Usability and User Feedback

- 4) **Ease of Use:** Most participants found the user interface clean and easy to navigate, with minimal onboarding required.
- 5) **Feature Satisfaction:** Users appreciated the automated expense splitting and real-time status display, highlighting reduced friction compared to manually dividing amounts in typical UPI apps.
- 6) **Improvement Suggestions:** Feedback pointed toward potential enhancements, such as richer notification options, improved device compatibility, and more granular permissions for group management.

D. Comparison with Related Solutions

- 1) **Feature Set:** Unlike most mainstream UPI applications, UPIshare natively supports group payments, historical tracking, and collaborative workflows, similar to global apps like Splitwise but fully integrated with the UPI backend.
- 2) **Open-Source Nature:** UPIshare's transparent and extendable codebase sets it apart from proprietary solutions, encouraging trust and community-driven development.

E. Limitations

- 1) **Current integration with UPI** is dependent on partner APIs and mobile app compatibility.
- 2) **Notification mechanisms** may vary in reliability across device ecosystems.
- 3) **Full automation of payment status** requires tight coupling with UPI infrastructure, which is subject to external limitations.

VI. DISCUSSION

The development and deployment of UPIshare provide valuable insights into both the technical and sociocultural aspects of collaborative digital payments in India. This section highlights the project's key achievements, elaborates on challenges faced during implementation, and summarizes critical lessons learned.

A. Key Achievements and Contributions

UPIshare successfully addresses a noticeable gap in the UPI ecosystem by offering integrated group payment workflows, an intuitive user interface, and transparent transaction management. Its open-source nature fosters adaptability, encouraging community contributions, deployment in varied contexts, and accelerated innovation compared to closed, proprietary alternatives. By leveraging robust API integrations and a modular architecture, UPIshare ensures security, maintainability, and future extensibility.

One prominent achievement is the facilitation of seamless group expense management—transforming what is often a tedious, manual process into an efficient, transparent workflow. This also empowers users with clear payment records and reduces the social friction often associated with settling shared accounts.

B. Challenges Faced

- 1) **Technical Integration:** One of the primary technical challenges was the integration with UPI's rapidly evolving APIs and handling compatibility with multiple UPI client applications and bank implementations. Ensuring reliable confirmation and reconciliation of payments across diverse platforms required sophisticated error handling and fallback mechanisms such as manual confirmation when automated callbacks were unavailable.
- 2) **Security and Privacy:** Given the sensitive nature of financial transactions, UPIshare's architecture had to prioritize data protection through encryption, secure coding practices, and strict access controls. Meeting regulatory requirements while supporting an open-source model added an additional layer of complexity.
- 3) **User Experience:** Designing a user interface that is both feature-rich and accessible—even for users with limited digital literacy—posed a significant challenge. Balancing advanced features with simplicity necessitated iterative usability testing and feedback-driven design refinements.
- 4) **Notification and Platform Diversity:** Delivering real-time notifications reliably across the diverse mobile and web ecosystem in India presented additional hurdles. Variability in notification delivery and OS-level restrictions impacted the consistency of user alerts.

C. Lessons Learned and Best Practices

The experience of building UPIshare underlines the importance of:

- 1) Rigorous modularization and interface segregation for scalability and maintainability.
- 2) Early adoption of secure-by-design principles, especially in the context of financial applications.
- 3) Incorporating user feedback continuously, ensuring that real-world needs and pain points drive feature prioritization.
- 4) Maintaining clear and thorough documentation to attract and support contributions from the open-source community.

The collaborative nature of open-source projects like UPIshare illustrates the potential for rapid improvement and widespread impact when transparency and collective expertise are leveraged.

VII. CONCLUSION AND FUTURE WORK

The UPIshare project demonstrates a meaningful advance in digital payment solutions by bridging the gap between the robust UPI infrastructure and the nuanced needs of collaborative expense management. Through its user-friendly interface, secure architecture, and transparent group workflows, UPIshare enables individuals and communities to manage shared payments seamlessly. Its open-source foundation not only encourages trust and code transparency but also cultivates an environment for innovation and rapid adaptation to evolving requirements.

The evaluation of UPIshare highlights its effectiveness in simplifying group financial interactions, improving payment tracking, and enhancing user experience compared to existing UPI applications. Its modular, extensible design, combined with strong privacy and security measures, makes it a valuable resource for both end-users and developers interested in digital finance.

However, some challenges and limitations remain. Efforts to provide deeper integration with the latest UPI APIs, ensure compatibility across an even broader spectrum of devices and payment apps, and optimize notification systems are ongoing. Addressing these will increase reliability and enable more sophisticated automation.

Future Work

Moving forward, UPIshare can evolve in several directions:

- 1) **Advanced Analytics:** Integrating analytics and reporting features to help users analyze spending patterns and group contributions.
- 2) **Enhanced Automation:** Further automating payment confirmation processes by utilizing emerging UPI protocols and real-time APIs.
- 3) **Mobile Application Development:** Expanding to native mobile platforms (Android/iOS) for richer notifications and offline access.
- 4) **Localization and Accessibility:** Supporting regional languages and accessibility features to widen inclusivity and adoption.
- 5) **Integration With Other Financial Tools:** Building connectors for budgeting, accounting, and tax management platforms.
- 6) **Community Expansion:** Engaging with the open-source community for contributions, plugin development, and third-party integrations.

In summary, UPIshare lays a strong technological foundation for collaborative digital finance. Its continued development and adoption can significantly influence group payment practices and contribute to the evolving landscape of cashless transactions.

REFERENCES

- [1] Vidani, Jignesh. "A study on the rise and recent development in unified payments interface." Available at SSRN 4849785 (2024).
- [2] Trinity, Galipothula Hanok, and Nidhi Sharma. "Unified Payments Interface (UPI): Fraud and Prevention Strategies." 2025 International Conference on Electronics, Computing, Communication and Control Technology (ICECCC). IEEE, 2025.
- [3] Cornelli, Giulio, et al. "The organisation of digital payments in India—lessons from the Unified Payments Interface (UPI)." SUERF: The European Money and Finance Forum, SUERF Policy Note. No. 355. 2024.
- [4] More, Ruchik Sunil. Understanding Factors Influencing UPI (Unified Payments Interface) user adoption levels and sentiments in India. Diss. Dublin, National College of Ireland, 2024.
- [5] Mahesh, A., and Ganesh Bhat. "Digital payment service in India—a case study of unified payment interface." International Journal of Case Studies in Business, IT and Education (IJCSBE) 5.1 (2021): 256-265.
- [6] Johari, Anikesh. Unified Payments Interface and Financial Inclusion in Rural India. Diss. Vrije Universiteit Brussels, Belgium, 2024.
- [7] Momjian, Bruce. PostgreSQL: introduction and concepts. Vol. 192. New York: Addison-Wesley, 2001.
- [8] Drake, Joshua D., and John C. Worsley. Practical PostgreSQL. "O'Reilly Media, Inc.", 2002.
- [9] Douglas, Korry, and Susan Douglas. PostgreSQL: a comprehensive guide to building, programming, and administering PostgreSQL databases. SAMS publishing, 2003.
- [10] Ravshanovich, Azizbek Rajabov. "Database structure: PostgreSQL database." Psixologiya va sotsiologiya ilmiy jurnali 2.7 (2024): 50-55.
- [11] Fedosejev, Artemij. React.js essentials. Packt Publishing Ltd, 2015.
- [12] Aggarwal, Sanchit. "Modern web-development using react.js." International Journal of Recent Research Aspects 5.1 (2018): 133-137.
- [13] Rajala, Ossian. "Impact of React component libraries on developer experience—An empirical study on component libraries' styling approaches." (2024).
- [14] Deshpande, Madhura. "An analytical study of use of UPI payment gateways among postgraduate students in PCMC." International Journal of Scientific Research in Engineering and Management 9 (2025): 1-9.
- [15] Ruban, Mr. R. Mohana. "A STUDY ON USER'S OPINION AND AWARENESS TOWARDS THE METHODS OF UPI-DIGITAL PAYMENT GATEWAYS (WITH SPECIAL REFERENCE TO THE USERS OF TIRUVALLUR DISTRICT)."
- [16] Charan, Gangisetty Raj, and K. Deepa Thilak. "Detection of phishing link and QR code of UPI transaction using machine learning." 2023 3rd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA). IEEE, 2023.
- [17] David, Surentheran, John Zoe Shalom, and S. Kumaran. "Mobile Wallet Adoption in India: Analyzing Consumer Behavior Through UPI and QR Code Usage." 2025 3rd International Conference on Advances in Computation, Communication and Information Technology (ICAICIT). Vol. 1. IEEE, 2025.
- [18] Sheth, Taneesha. "Beyond the QR Code: The Limits of UPI-Led Financial Inclusion and the Systems That." (2026).
- [19] Leith, Douglas J. "Web Browser Privacy: What Do Browsers Say When They Phone Home?." IEEE access 9 (2021): 41615-41627.
- [20] Horsman, Graeme, et al. "A forensic examination of web browser privacy-modes."
- [21] Forensic Science International: Reports 1 (2019): 100036.
- [22] Tandel, Sayali, and Abhishek Jamadar. "Impact of progressive web apps on web app development." International Journal of Innovative Research in Science, Engineering and Technology 7.9 (2018): 9439-9444.
- [23] Hume, Dean. Progressive web apps. Simon and Schuster, 2017.
- [24] Malavolta, Ivano, et al. "Assessing the impact of service workers on the energy efficiency of progressive web apps." 2017 IEEE/ACM 4th International Conference on Mobile Software Engineering and Systems (MOBILESoft). IEEE, 2017.
- [25] Ezhilchelvan, Paul, and I. S. Mitran. "Static and dynamic hosting of cloud servers." European Workshop on Performance Engineering. Cham: Springer International Publishing, 2015.
- [26] Pyrgelis, Apostolos, Emiliano De Cristofaro, and Gordon J. Ross. "Privacy-friendly mobility analytics using aggregate location data." Proceedings of the 24th ACM SIGSPATIAL international conference on advances in geographic information systems. 2016.
- [27] Neema, Kratika, and Arpit Neema. "UPI (Unified Payment Interface)—a new technique of digital payment: an explorative study." International Journal of Current Research in Multidisciplinary 3.10 (2016): 1-10.



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)